

Benchmarking KAZE and MCM for Multiclass Classification

Siddharth Srivastava, Prerana Mukherjee, and Brejesh Lall

Indian Institute of Technology, Delhi, India
 {eez127506, eez138300, brejesh}@ee.iitd.ac.in

Abstract. In this paper, we propose a novel approach for feature generation by appropriately fusing KAZE and SIFT features. We then use this feature set along with Minimal Complexity Machine(MCM) for object classification. We show that KAZE and SIFT features are complementary. Experimental results indicate that an elementary integration of these techniques can outperform the state-of-the-art approaches.

Keywords: Object Classification, KAZE, SIFT, Minimum Complexity Machine

1 Introduction

Majority of computer vision research e.g. in the area of image classification, object recognition, localization, detection, segmentation, retrieval etc. involves objects in one form or the other. Development of algorithms in these areas is driven by trade-off between robustness and scalability. In this paper we focus on the very challenging problem of object classification. In recent times, machine learning techniques have found favour among researchers addressing the image classification problem. Researchers are revisiting the deep learning tools and taking permutations of feature sets to encompass the various characteristics of the images. The diverse nature of objects make it difficult to devise a single solution to handle all object classification problems. The challenges in this area of research can be attributed to the following factors:

- Number of classes
- Number of instances of each class
- Total number of images in the dataset
- Relative ratios of training and testing images
- Intra-class variance due to clutter, pose variations, occlusion, illumination changes etc.
- Ground truth annotations.

These factors have an impact not only on the efficiency of the technique but also upon its complexity. Having a large number of classes poses the challenge of capturing the intra-class as well as inter-class variation precisely. The size of training data as well as the variation in it decides has a direct impact on

the discriminability (and hence complexity) of the features chosen. On one hand, large data is good for training the classifier however, on the other hand it leads to the requirement of complex features. Also training with a small dataset results in the problem that it does not capture the variational changes, whereas, a larger dataset makes annotation difficult [17].

Various techniques attempt to address one or more of these factors for object classification. Broadly, the pipeline of such solutions may be described as in Figure 1.

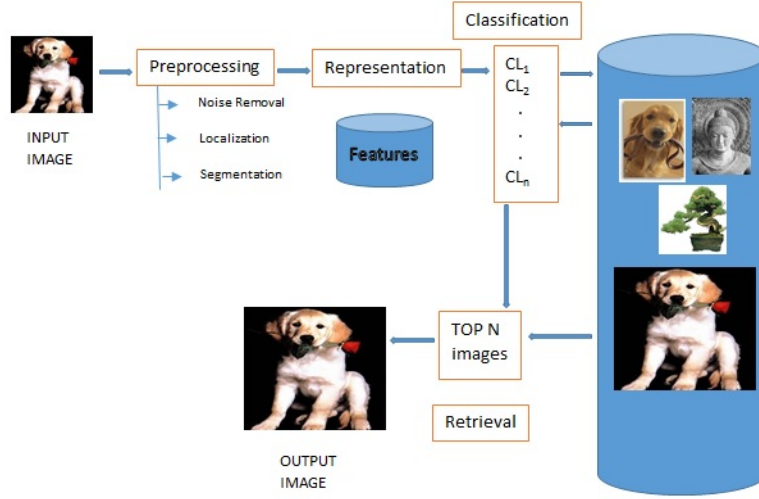


Fig. 1. Object Classification Pipeline

Popular approaches for object classification can be categorized as follows:

1. Techniques which focus primarily on improving the input representation with the help of stronger features while using a simple classifier such as SVM. Sc-SPM proposed in [23] chose sparse coding over vector quantization. Accordingly, it relaxes the cardinality constraints and introduces a regularization parameter to obtain a smaller number of non zero elements. This is then followed by max spatial pooling reducing the complexity of the classifier. In [22], authors use a locality adaptor which allows to choose appropriate basis vector corresponding to an input descriptor.
2. Techniques focusing on using classifiers by generating stronger training cases as compared to the approaches briefed in 1. In [8], authors formulate a latent SVM which results in the problem being formulated as a convex training problem. They also propose a HoG like feature descriptor which is also used to generate hard training examples for building a stronger classifier. In [9], the authors introduce R-CNN, a variant of convolutional neural network

to extract features from the region proposals which are then classified into respective object categories.

3. Techniques which try to balance the trade-off between speed and accuracy by tuning both 1 and 2. In [11], authors propose a two stage sliding window approach for object localization. The main idea is to combine the classification and detection phases by considering latent properties of objects and scenes. Another technique, Selective Search [19] reduces the relative time for localizing objects, hence allowing for stronger classification techniques.

As is evident from the previous discussion, the choice of features and classifier play a crucial role in the quality of the object classification techniques. Despite this strong dependence on choice of features, SIFT [14] and its variants [20] [13] [15] have remained the de-facto choice for feature representation. SIFT is based on Gaussian scale space (GSS) which blurs the image uniformly, resulting in loss of distinctness in the object boundaries. Recently, a work proposes to use non linear scale space, which preserves the object boundaries by blurring the region around edges more than the edges themselves. KAZE [1], which is based on the non-linear scale space, hence is a promising choice for the features for object classification. KAZE features have strong responses around object boundaries, while SIFT features capture the details (at the boundary or otherwise) in an image. According to [2], an object can be characterized by a well defined boundary, a distinctive appearance and a salient region. Therefore, a feature set comprising of carefully chosen KAZE and SIFT keypoints is an appropriate choice for defining an object.

For the other key operations in object classification, Support Vector Machines (SVM) [6] have been the traditional choice. Recently, Minimal Complexity Machine (MCM) [12] has shown to outperform SVM in terms accuracy, computational complexity as well as sparse representation of the features. The strongest argument in favor of MCM is its provably good generalization accuracy and requirement of far less number of support vectors as compared to SVMs. Fewer support vector mean faster classification of test points. Due to complexity and size of the object classification datasets, MCM makes a strong case for itself. A more detailed discussion about MCM as compared to SVM is given in Section 2.

In light of the discussions above, we present a novel technique for generating a stronger feature set by careful combination of KAZE and SIFT keypoints (SIFT-KAZE). We use these features with MCM to propose a light weight but stronger object classifier. The contributions of this paper can be summarized as follows:

1. This paper establishes that SIFT and KAZE are complementary features and a tuned combination of these is better suited for object classification tasks.
2. This is the first work to demonstrate the effectiveness of MCM on images and datasets with large number of classes. Further, the proposed technique outperforms the traditional methods by a significant margin and can be easily integrated with the existing techniques. This can lead to the development of more efficient yet simpler techniques in this domain.

Rest of the paper is organized as follows: In Section 2, we introduce the fundamental analysis of the non linear scale space and demonstrate its effectiveness in combination to the object boundary representation and go on to propose the object classification technique. Section 3, presents the experimental analysis. Section 4, elaborates the results which were obtained. Section 5, concludes the paper.

2 Discriminant Keypoint based Classifier

2.1 Beyond SIFT

The major difference between KAZE and SIFT is in the construction of the scale space. KAZE is based on non-linear scale space while SIFT is based upon Gaussian scale space(GSS). KAZE uses non-linear diffusion filtering. This diffusion process is formulated in equation (1)

$$\frac{\partial L}{\partial t} = \text{div} \{ (c(x, y, t) \cdot \nabla(L)) \} \quad (1)$$

where div and ∇ are divergence and gradient operators, c is the conductivity function and t is scale parameter. The conductivity function c , is represented as a gradient(Equation (2)), helping in the reduction of diffusion at edges resulting in more smoothening of regions than edges. This property of the conductivity function makes it more suitable for boundary representation.

SIFT constructs GSS which blurs both the object region and boundary. This helps in characterization of object using high detail interest points(not necessarily boundary points). KAZE uses the general diffusion equation as shown in equation (2) to construct the scale space. There are various conductivity functions defined in [16], which can be used to promote high contrast, wider regions or smoothening on both sides of the edges.

$$c(x, y, t) = g(|\nabla L_\sigma(x, y, t)|) \quad (2)$$

In SIFT, the base image for each octave is generated by downsampling the image from previous octave whereas in KAZE, the construction of each octave is based on the original image .

We now define two measures Keypoint Overlap Score (KOS) and Mean Keypoint Overlap Score (MKOS) to evaluate the effectiveness of SIFT and KAZE in providing discriminative keypoints.

The Keypoint Overlap Score(Equation 3) of an image I is defined as the percentage of the number of keypoints within the ground truth bounding boxes BB_o for each object o in the image.

$$KOS = \frac{1}{K} \left[\sum_{o=1}^O \sum_{k=1}^K \chi(BB_o, KP_k) \right] \quad (3)$$

where O is the number of objects in the image, K is the total number of keypoints

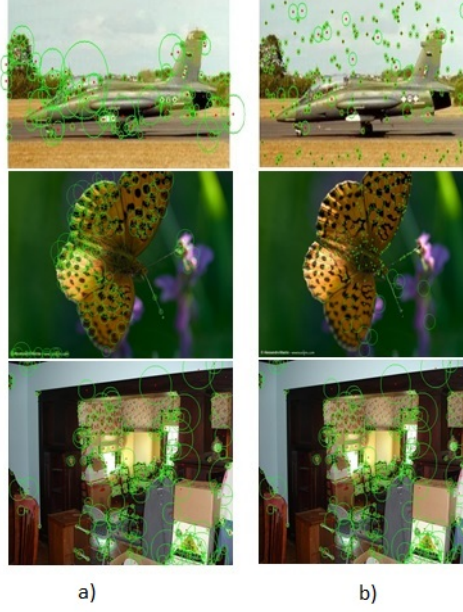


Fig. 2. a) Shows the KAZE keypoints which are densely distributed along the object boundaries and b) Shows the SIFT keypoints around the regions.

detected in the image, BB_o is the bounding box of object o , KP_k is the k^{th} keypoint and $\chi(BB_o, KP_k)$ specifies if a keypoint KP_k lies within the bounding box BB_o (Equation 4).

$$\chi(BB_o, KP_k) = \begin{cases} 1 & \text{if } KP_k \text{ within } BB_o \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

$$MKOS = \frac{1}{N} \sum_{i=1}^N KOS_i \quad (5)$$

where KOS_i is the Keypoint Overlap Score for Image i and N is the total number of images

Since KOS is image specific, we define a generic goodness measure MKOS as the average over all the images considered for evaluation (Equation 5).

The KOS and MKOS are calculated on Pascal VOC 2007 [7] dataset. To characterize the boundaries from the ground truth annotations, we also create a region around the ground truth box BB_o by extending and reducing it with a factor of β as shown in equations (6) and (7). The scores are then calculated for the region represented by A_{region} .

$$A_{extended}\{BB_o\} = A_{original}\{BB_o\} * (1 + \beta) \quad (6)$$

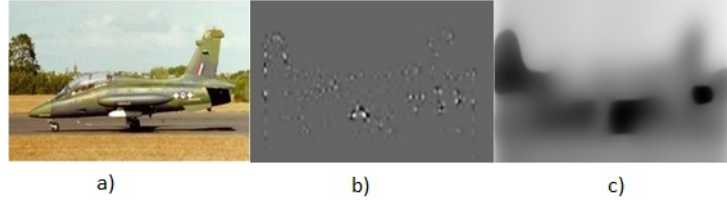


Fig. 3. a) Original Image b) KAZE detector response localised around the object (aeroplane) c) KAZE scalespace

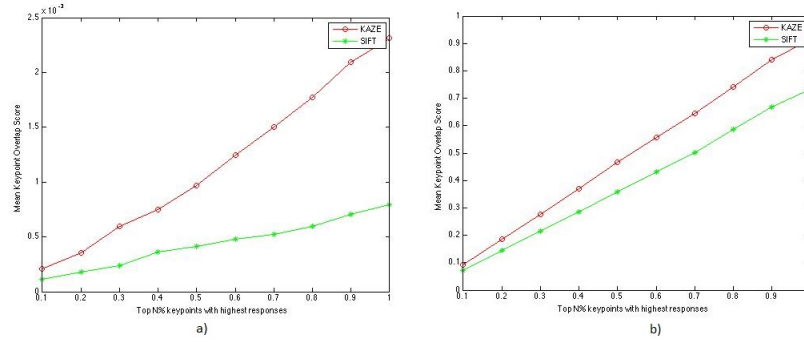


Fig. 4. (a) Mean Average Keypoint Overlap Score vs Top N% keypoints with highest responses (For all keypoints within the bounding box) (b) Mean Average Keypoint Overlap Score vs Top N% keypoints with highest responses (For all keypoints within $\beta = 0.1$)

$$A_{reduced} \{BB_o\} = A_{original} \{BB_o\} * (1 - \beta) \quad (7)$$

$$A_{region} \{BB_o\} = A_{extended} \{BB_o\} \cap A_{reduced} \{BB_o\} \quad (8)$$

Figure 4(a) and (b) shows the MKOS score for this dataset. The results were produced by calculating KAZE and SIFT keypoints and responses for each image in the dataset and sorting them according to the responses of the keypoints. The MKOS was then calculated for top N% of the keypoints. As it is shown, KAZE consistently outperforms SIFT with the density of KAZE features being heavily concentrated around the object boundaries. Though representation of object boundary is an important factor for object classification, it is not the sole discriminating factor which follows from the experimental analysis in Section 3. Analytically non-linear scale space preserves edges and hence it is not surprising that most of the KAZE keypoints are concentrated at the boundary. SIFT on the other hand looks for sharp discontinuities at all scales and can hence capture keypoints inside a region. This phenomena can also be observed visually from Figure 2, SIFT gives a high number of keypoints in relatively less salient regions (like grass, clouds etc.), while KAZE features were dominant around the most

salient region boundaries (i.e. the object boundaries). It can be observed in Figure 3 that KAZE does not blur out the object in the detector response as well as scale space.

2.2 SVM vs MCM

Minimal Complexity Machine is based on bounding the Vapnik-Chervonenkis (VC) dimension. VC dimension is a measure to establish the effectiveness of a machine learning algorithm. Alternatively, consider a parametric model $M(\alpha)$ and a set of data points \mathbf{X} . Now if there exists a parameter α for model M , such that all possible label assignments \mathbf{L} to \mathbf{X} are classified without errors. The model $M(\alpha)$ is then said to have shattered the data points \mathbf{X} . The largest number of data points that can be shattered by $M(\alpha)$ is defined as the VC dimension of this model. Thus, VC dimension gives a family of functions that separates the input set of points. More intuitively, VC dimension therefore sets an upper bound on the test error rate[21]. The performance of the model is evaluated by risk associated with it, which is given as

$$Risk \leq EmpiricalRisk + f(h) \quad (9)$$

where h is the VC dimension.

Here, the empirical risk is the classification error rate while f is a monotonically increasing function.

Now as noted by Burges[4], SVMs may have a very high VC dimension. Consequently, it would have a high risk associated it as compared to a model with the same classification error rate. In contrast, MCM guarantees good generalization accuracy by obtaining a better bound (lower and upper) over the VC dimension while also achieving excellent training error rates. In addition as noted in [12], the number of support vectors obtained by MCM are comparatively less than that of SVM. This makes MCM suitable for complex classification tasks while also providing opportunity to reduce the overall overhead during classification. Since MCM solves a linear programming problem, it provides a significant performance gain over the quadratic programming problem solved by SVM.

3 Experiments and Results

In the experiments we evaluated SIFT, KAZE and the proposed SIFT-KAZE against each other using SVM and MCM as classifiers. We have provided an exhaustive evaluation over Caltech-256 dataset[10].

We represent the features as bag of visual words which is then provided to SVM and MCM for further classification. For multiclass classification, We have used the one-vs-one approach and libSVM[5] implementation for SVM classification. As the patterns represented by SIFT features are linearly separable [23], a linear kernel is more suitable for classification. Therefore, for our experiments, we have used linear kernel instead of a non-linear kernel. We have found that

the patterns represented by KAZE features are also linearly separable, since our experiments with a non linear(RBF) kernel were consistently performed worse than those with linear kernel.

The results are shown in Tables 1,2 and 3. MCM outperforms SVM in all the experiments. Here, it is important to reiterate that the contemporary works achieving state of the art performance using SVM used strong pre-processing techniques or were trained with specifically constructed hard negatives from the training examples whereas in this work, we have used the simplest representation of features and classifiers.

The comparatively lower classification accuracy of KAZE can be attributed to the fact that KAZE features have a high density along the boundary of the objects. This establishes that despite the fact that boundary is the strongest distinguishing property of an object, it is not the definitive criteria [2]. On the contrary, the weighted mixture of SIFT and KAZE (Table 3) outperform the other two approximately by 2-3% for MCM and around 8%-10% for SVM. This strengthens the claim that SIFT and KAZE are complementary features and can strongly define an object within an image. This can be understood by observing the fact that while KAZE effectively incorporates the boundary characteristics, SIFT prominently captures the region properties.

Table 4, shows the performance of the state of the art technique on Caltech-256 dataset. As can be seen that only the CNN using ImageNet(pretrained) [24] outperforms our method. It is important to note that despite using the most basic technique, we were able to outperform many advanced and relatively complex techniques while also achieving comparable results to the state of the art. This is a key observation since the presented set of techniques are generic and hence numerous variants may be derived similar to contemporary techniques utilizing SVM and SIFT with myriad kind of tunings, preprocessing etc.

Table 1. Classification accuracy for MCM and SVM for SIFT features on Caltech-256 dataset.

Training Samples	MCM	SVM
15	52.79	19.82
30	55.08	26.82
45	56.45	28.98
60	57.20	30.91

Table 2. Classification accuracy for MCM and SVM for KAZE features on Caltech-256 dataset.

Training Samples	MCM	SVM
15	51.83	18.24
30	52.00	21.08
45	52.70	22.86
60	52.90	24.23

Table 3. Classification accuracy for MCM and SVM for Mixture of SIFT and KAZE features on Caltech-256 dataset.

Training Samples	MCM	SVM
15	56.93	26.86
30	57.13	34.92
45	58.68	38.95
60	59.66	42.60

Table 4. State of the art classification accuracy on Caltech-256

Technique	15	30	45	60
ScSPM[2009][23]	27.73	34.02	37.46	40.14
LLC[2010] [22]	34.36	41.19	45.31	47.68
Multipath Sparse Coding[2012] [3]	40.5	48.0	51.9	55.20
SIFT+Fisher Vector[2013][18]	38.5	47.4	52.1	54.8
SIFT+LCS+Fisher Vector[2013][18]	41.0	49.4	54.3	57.3
CNN using ImageNet pretrained[2014][24]	65.7	70.6	72.7	74.2

4 Conclusion and Future Scope

We have established that SIFT and KAZE features represent complementary information of an object and a fusion of these techniques along with MCM outperforms the state of the art, while achieving remarkable improvement over SVM classification. We also evaluated the effectiveness of MCM for image datasets. The set of techniques used in this paper are simple yet powerful, we trust that they have the potential to significantly improve the more sophisticated (complex) state of the art techniques.

References

1. P. F. Alcantarilla, A. Bartoli, and A. J. Davison. Kaze features. In *Computer Vision–ECCV 2012*, pages 214–227. Springer, 2012.
2. B. Alexe, T. Deselaers, and V. Ferrari. What is an object? In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 73–80. IEEE, 2010.
3. L. Bo, X. Ren, and D. Fox. Multipath sparse coding using hierarchical matching pursuit. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 660–667. IEEE, 2013.
4. C. J. Burges. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2):121–167, 1998.
5. C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
6. C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.
7. M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. <http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html>.
8. P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645, 2010.
9. R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 580–587. IEEE, 2014.
10. G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. 2007.
11. H. Harzallah, F. Jurie, and C. Schmid. Combining efficient object localization and image classification. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 237–244. IEEE, 2009.
12. Jayadeva. Learning a hyperplane classifier by minimizing an exact bound on the $\{VC\}$ dimension1. *Neurocomputing*, 149, Part B(0):683 – 689, 2015.
13. Y. Ke and R. Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–506. IEEE, 2004.
14. D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

15. E. N. Mortensen, H. Deng, and L. Shapiro. A sift descriptor with global context. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 184–190. IEEE, 2005.
16. P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 12(7):629–639, 1990.
17. N. Pinto, D. D. Cox, and J. J. DiCarlo. Why is real-world visual object recognition hard? *PLoS computational biology*, 4(1):e27, 2008.
18. J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek. Image classification with the fisher vector: Theory and practice. *International journal of computer vision*, 105(3):222–245, 2013.
19. J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013.
20. K. E. Van De Sande, T. Gevers, and C. G. Snoek. Evaluating color descriptors for object and scene recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1582–1596, 2010.
21. V. N. Vapnik and V. Vapnik. *Statistical learning theory*, volume 1. Wiley New York, 1998.
22. J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3360–3367. IEEE, 2010.
23. J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1794–1801. IEEE, 2009.
24. M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *Computer Vision–ECCV 2014*, pages 818–833. Springer, 2014.